



# Scheduling transmissions with latency constraints in an IEEE 802.15.4e TSCH network

Ines Khoufi, Pascale Minet, Badr Rmili

## ► To cite this version:

Ines Khoufi, Pascale Minet, Badr Rmili. Scheduling transmissions with latency constraints in an IEEE 802.15.4e TSCH network. VTC 2017 - IEEE 86th Vehicular Technology Conference, Sep 2017, Toronto, Canada. hal-01636656

**HAL Id: hal-01636656**

**<https://hal.archives-ouvertes.fr/hal-01636656>**

Submitted on 16 Nov 2017

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Scheduling transmissions with latency constraints in an IEEE 802.15.4e TSCH network

Ines Khoufi\*, Pascale Minet\* and Badr Rmili†

\*Inria, 2 rue Simone Iff, CS 42112, 75589 Paris Cedex 12, France. Email: firstname.name@inria.fr

†CNES Launcher Directorate, 52 rue Jacques Hillairet, 75012 Paris, France. Email: badr.rmili@cnes.fr

**Abstract**—Time Slotted Channel Hopping (TSCH), specified in the IEEE 802.15.4e amendment, has been designed for industrial automation, process control and equipment monitoring. It uses a slotted medium access on several channels in parallel and supports multihop communications. In this paper, we study how applications with data delivery constraints can be supported by a TSCH network. We first propose a framework based on a multislotframe that allows the coexistence of Data Slotframes and Control Slotframes. We then determine a lower bound on the minimum number of slots required to perform data gathering, taking into account the number of channels, the number of interfaces of the sink, the number of packets generated by each sensor node as well as the number of children of the sink. These feasibility conditions are established for two cases: with spatial reuse and without. We propose a debt-based scheduler that for simple topologies, provides a schedule minimizing the slotframe size. Finally, we consider a network configuration representative of an industrial application and evaluate the performance of the TSCH network in terms of data delivery delay and queue size for each sensor node, using the NS-3 simulator. Simulation results confirm the theory.

## I. CONTEXT AND MOTIVATION

Most wireless sensor networks (WSNs) deployed up to now use a technology based on the IEEE 802.15.4 standard. This standard has been recently amended with the IEEE 802.15.4e amendment to meet the requirements of industrial applications with regard to latency, robustness and energy. Among the five modes proposed, the TSCH (Time Slotted Channel Hopping) mode has been designed for industrial automation, process control and equipment monitoring. The TSCH network should ensure that the time needed to gather all the data generated by sensor nodes is less than a given threshold, usually called delivery latency. The TSCH network should be able to tolerate some channel perturbations. These perturbations can be caused either on the one hand by devices belonging to the same network or another network coexisting in the same geographical area, or on the other hand by an external source (e.g. a radar). In addition, since some sensor nodes are battery-equipped, their lifetime should be maximized. Huge energy savings can be achieved by making sensor nodes sleep. However, this should not degrade the performances of data gathering. The TSCH solution is based on a multichannel time-slotted medium access using channel hopping. It supports star, tree and mesh topologies. These features make it a very promising candidate for industrial applications. That is why TSCH is the focus of this paper.

Most of the studies on IEEE 802.15.4e TSCH networks focus on communications in an operational network. Some

evaluate the average throughput, the average delays and the average energy consumption either by means of simulations like [1], or analytical models like Markov models in [2], or estimators validated on real implementations like [3] and [4]. Others compute upper bounds on the delays for star topologies, using Network Calculus like [5]. The gains brought by TSCH networks with regard to IEEE 802.15.4 networks are highlighted in [1], where the NS2 simulator is used to compare the performances of both networks for a star topology with a number of nodes ranging from 20 to 120 in terms of delivery ratio, energy consumed and delivery delays. TSCH outperforms classical 802.15.4 both in beacon-enabled and non-beacon-enabled modes. In [2], the authors model the transmission delays obtained with the shared slots of TSCH, for which collisions may occur and compare the performance with CSMA/CA, pointing out the differences in collision avoidance used by TSCH and CSMA/CA.

The contribution of this paper is twice. First we determine theoretical bounds on the minimum number of slots needed by data gathering and on the maximum delivery time, taking into account network parameters. Second, we focus on an industrial application with strong latency constraints and evaluate the TSCH performances in various configurations with the NS3 simulator. This paper is organized as follows: Section II presents the theoretical framework of this study and defines the problem. In Section III, the feasibility conditions are given for a valid schedule to meet the latency constraint, taking into account the topology, the sampling rate of each sensor node, the data gathering tree, the number of interfaces of each wireless node (sensor node or sink), the number of channels simultaneously used by the WSN. Section IV proposes a debt-based scheduler to build a valid schedule meeting the latency constraint, if the problem is feasible. Section V reports the simulation results obtained with the NS3 simulation tool for various configurations (random topologies, heterogeneous sampling rates, number of interfaces of the sink). Simulation results confirm that the schedule provides the minimum number of slots required by the configurations studied.

## II. THEORETICAL FRAMEWORK

TSCH uses time-synchronized communications (time slots) and different channels in parallel (with frequency hopping) to improve network robustness by a spectral and time redundancy. The minimization of collisions increases the through-

put. Channel hopping limits the effects of multipath and interferences. TSCH can be used with any network topology.

There exists 16 frequencies available in the 2.4 GHz frequency band according to the IEEE 802.15.4 [6] standard that rules the physical layer (PHY) of any TSCH network. In this study, a constraint of having at most three TSCH networks present in the same area is considered. Thus, to avoid inter-networks interferences, the number of frequencies available per network is  $N_{Channel} \leq 5$ . Concerning the constraint on the distance between the frequencies used by a multi-interface node, usually the sink, it is preferable to use frequencies that are not direct neighbors. For instance, the set of frequencies used by a network could be a subset of 5 frequencies such as  $\{11, 15, 19, 23, 26\}$ .

#### A. MultiSlotframe

The solution that we propose to meet the latency and robustness constraints is based on the concept of MultiSlotframe. A MultiSlotframe is an ordered sequence of Slotframes that is periodically repeated. Figure 1 depicts an example of a simple MultiSlotframe.



Fig. 1: An example of a MultiSlotframe.

All the Slotframes of a same MultiSlotframe have the same size expressed as a number of slots, but they differ by their type. We define three different types of Slotframe:

- *Beacon*: this type is used for the downstream cascading transmission (i.e. from the sink to the sensor nodes) of control frames, the Enhanced Beacons. By means of Information Elements (IE) contained in the Beacon specified by the IEEE 802.15.4e standard [7] or created to meet our needs, this control traffic performs neighborhood discovery, selection of potential parents (routing in the data gathering tree) and the installation of schedules used for data gathering. A multihop cascading according to the address of each node allows each node to receive the information broadcast by the sink (i.e. the CPAN).
- *Shared*: this type is used for the transmission of downstream / upstream control traffic during the association of nodes (insertion of a node in the network) and the notifications (e.g. application or topology change detected by a node). Collisions are possible.
- *Data*: this type allows the transmission of useful data frames, in TDMA mode, without collisions.

The positioning of these different types of Slotframes has a very strong impact on the latency that is met by data

gathering on the one hand, and on the buffer size on sensor nodes on the other hand. To avoid that data accumulate on the nodes while waiting for their transmission, it is highly recommended that the Data Slotframes are fairly distributed in the Multislotframe to minimize the value of  $Reprod$  that denotes the maximum number of consecutive Slotframes to consider in the MultiSlotframe to get one Data Slotframe. In the example depicted in Figure 1, the Data Slotframe is repeated at least every 2 Slotframes. Hence  $Reprod = 2$  in this example. Notice that if all Slotframes are Data Slotframes,  $Reprod = 1$ .

#### B. Slotframe

The following property ensures that the transmission of a node  $N_1$  to a neighboring node  $N_2$ , scheduled in the SlotOffset  $j$  and done on the channel  $ch_1$  in the Slotframe  $i$  will be done in a channel  $ch_2 \neq ch_1$  in the Slotframe  $i + 1$ . This rule guarantees that all channels of the TSCH network are visited by the transmissions of  $N_1$  to  $N_2$  scheduled in SlotOffset  $j$  in the successive Slotframes.

*Property 1:* For robustness reasons, the size of the Slotframe, expressed as a number of slots, and the number of channels used by the network should be co-prime.

If this property is not met, any transmission scheduled in a given SlotOffset will only visit a subset of the channels granted to the TSCH network, which does not allow the TSCH network to take full advantage of frequency diversity. The size of the Slotframe is selected to meet this property. Its value is advertised in the Enhanced Beacons.

The MAC layer is in charge of managing multichannel medium accesses according to the Slotframe. Two types of slots are distinguished. *Shared slots* provide a medium access in CSMA/CA mode, with possible collision. *Dedicated slots* provide a medium access in TDMA mode, without collision.

A *Data Slotframe* has all its slots of type dedicated: only the transmitter is in Transmit state and only the receiver(s) specified are in the Receive state. A *Shared Slotframe* has all its slots of type shared: any node may transmit. A *Beacon Slotframe* has all its slots of type dedicated: in each slot, only one node is allowed to transmit (all nodes transmit in sequence, the sequence is determined by the nodes addresses). In each slot of a Beacon Slotframe, all nodes are in Receive state. The assignment of slots and channels is performed by the sink (i.e. the CPAN) which is in charge of computing the schedule.

#### C. Notation

In the following, we adopt the notations given in Table I for the TSCH network considered.

#### D. Dimensioning of the Slotframe and the MultiSlotframe

The size of the Slotframe and the MultiSlotframe must meet some constraints. These constraints are related to data

TABLE I: Notations.

$L$	maximum end-to-end delivery time acceptable by the application
$NChannel$	number of channels in the frequency hopping sequence, $NChannel > 1$
$NInterf$	number of interfaces of the sink
$NChild$	number of children of the sink
$Nnode$	number of nodes in the TSCH network
$Gen(i)$	number of packets locally generated by node $i$
$Reprod$	the Data Slotframe is repeated every $Reprod$ Slotframes, $Reprod = 2$ by default
$SlotframeSize$	number of slots in the Slotframe
$BI$	Beacon Interval, period between two successive advertisements by the same node
$NSlotframe$	number of slotframes in the MultiSlotframe
$SlotDuration$	duration of the timeslot = 10ms by default

gathering, beacon advertisement as well as control messages. Concerning the Beacon Slotframe, we have:

- Each node must be able to transmit its beacon once per MultiSlotframe. Hence the Slotframe size should meet:

$$Nnode \leq SlotframeSize \times NBeaconSlotframe \quad (1)$$

For a WSN with 20 sensor nodes and a sink, with 2 Beacon Slotframes per MultiSlotframe, we get  $SlotframeSize \geq 11$ .

- For robustness reason,  $SlotframeSize \times NSlotframe$  and  $NChannel$  should be coprime.
- The periodicity of beacon also called Beacon Interval, denoted  $BI$ , is given by

$$BI = NSlotframe \times SlotframeSize \times SlotDuration \quad (2)$$

For a MultiSlotframe with 15 Slotframes of 13 slots each, we get  $BI = 1.95s$ .

#### E. Problem statement

In the absence of link and node failure, the strongest constraint that must be met by data gathering concerns the maximum end-to-end delivery time, that should be less than or equal to a given *Latency*. To meet this constraint, data gathering must be scheduled in a smart way.

**The scheduling problem consists in minimizing the number of slots required to gather data from sensor nodes in a TSCH network, under the following constraints:**

- A0. The maximum end-to-end delivery time should be less than or equal to a given *Latency*.
- A1. Packets are timestamped when generated at the Network layer.
- A2. Each node transmits the packet with the smallest timestamp in its Transmit queue. This packet may be either a packet generated locally, or a packet received that has to be forwarded.
- A3. No link/node failures are assumed in this paper.
- A4. There is no aggregation done by intermediate nodes. Such data gathering is also called raw data convergecast.
- A5. Packets are only transmitted in Data Slotframes. Each packet is transmitted in a single time slot and

acknowledged in the same time slot (i.e. immediate acknowledgment).

- A6. The Data Slotframe is repeated every  $Reprod$  Slotframes.
- A7. Any data packet present at the beginning of the Data Slotframe is transmitted in this Slotframe and delivered to the sink in the same Slotframe.

### III. FEASIBILITY CONDITIONS

#### A. Sampling rate and Packetization

The first question that has to be answered when computing delivery delays of packets containing samples is how these packets are generated. Each packet is timestamped when it is generated at the network layer. The application determines the sampling period and the generation time of the first packet.

For the Packet Header at the Application level, we propose a format on 13 bytes: 8 bytes for the long address of the sensor node, 1 byte for the number of samples included in the packet, and 4 bytes for the packet timestamp. The useful size of a packet at the Application level is determined in Table II:

TABLE II: Useful size of a packet at the Application level.

Maximum size of a MAC frame	127 bytes
MAC Header + Trailer	11 bytes
Network Encapsulation	4 bytes
Packet Timestamp	4 bytes
Sample size	2 bytes
Application Packet Header	13 bytes
Useful size for Data	99 bytes

#### B. Slots needed by a raw data convergecast

With these notations, we can prove the following properties.

*Property 2:* In a raw data convergecast we have:

$$Number\ of\ packets\ received\ by\ the\ sink = \sum_{i \in WSN} Gen(i). \quad (3)$$

$$Total\ number\ of\ transmissions = \sum_{i \in WSN} Gen(i) * Depth(i). \quad (4)$$

*Proof :* By definition of raw data gathering, the sink must receive all the packets generated by each sensor node; hence, Equation 3. A packet generated by a node  $i$  at depth  $Depth(i)$  in the data gathering tree is transmitted  $Depth(i)$  times to reach the sink. Hence, the total number of transmissions is given by Equation 4.

Equation 4 highlights the impact of the data gathering tree on the number of transmissions that must be scheduled to perform data gathering.

*Property 3:* When all nodes in the wireless sensor network are in range of each other, the data gathering tree minimizing the latency is given by a binomial tree.

*Proof :* See [8].

This property has a limited interest since it considers a very specific topology where all nodes are in range of each other.

The problem being NP-hard, the extension to a multihop network is complex.

The number of slots needed strongly depends on the data gathering tree and on the traffic demand.

*Property 4:* In a raw data convergecast, the minimum number of slots assigned to sensor nodes is lower bounded by  $\max(S_n, S_t)$ , with:

$$S_n = \lceil \sum_{i \in WSN} Gen(i)/g \rceil + \delta_n \quad (5)$$

$$S_t = Gen(c_1) + 2 \sum_{\substack{v \in subtree(c_1) \\ v \neq c_1}} Gen(v) + \delta_t \quad (6)$$

with  $g = \min(NInterf, NChild, NChannel)$ , where  $NInterf$  denotes the number of radio interfaces of the sink,  $NChild$  the number of children of the sink,  $NChannel > 1$  the number of available channels for the convergecast.

$\delta_n = 1$  if  $\sum_{i \in WSN} Gen(i)$  is a multiple of  $g$  and  $Depth > 1$  and  $Nchannel \leq \min(Nchild, Ninterf)$ ,  $\delta_n = 0$  otherwise.  $\delta_t = 1$  if the  $(g+1)^{th}$  child requests the same number of slots as the first one, denoted  $c_1$ , the children of the sink being sorted by decreasing order of slot demands,  $subtree(c_1)$  is the subtree rooted at  $c_1$ ,  $\delta_t = 0$  otherwise.

*Proof :* The value of  $S_n$  expresses the fact that the sink must be able to receive all the packets generated by the sensor nodes that is  $\sum_i Gen(i)$ . Since the sink is able to receive at most  $\min(NChannel, NInterf, NChild)$  packets simultaneously, we get the value of  $S_n$ . The term  $\delta_n$  is introduced for topologies with a depth strictly higher than 1, when the number of messages received by the sink is a multiple of  $g$ . In this case, no node at depth 2 can transmit simultaneously with a child of the sink without causing a collision; an additional slot is needed for that purpose, hence the term  $\delta_n$ .

The value of  $S_t$  expresses the fact that the slot demand of the most demanding child of the sink must be met. Let  $c_1$  denote this child of the sink: it requires  $Gen(c_1)$  slots to transmit the packets locally generated, plus  $\sum_{\substack{v \in subtree(c_1) \\ v \neq c_1}} Gen(v)$  slots to receive the packets generated by its descendants, plus the same amount of slots to forward these packets. The term  $\delta_t$  is introduced for topologies where the  $(g+1)^{th}$  child of the sink requests the same number of slots as the first one, assuming that the children of the sink are ordered by decreasing slot demand. In this case, the first slot granted to the  $(g+1)^{th}$  child is one slot later than the first child. Since these two children have the same slot demand, the last slot granted to the  $(g+1)^{th}$  child is one slot later than the last slot of  $c_1$ .

Property 4 is very important for the following reasons: It supports sensor nodes with heterogeneous sampling rates. It adapts to various numbers of available channels for the network. It takes into account the number of interfaces of the sink. It applies to any topology. However, we can notice that the lower bound on the minimum number of slots is reached when there is no topology link that exists in addition to those

of the data gathering tree. From Property 4 we can also deduce that:

- It is useless to equip the sink with a number of radio interfaces strictly higher than the number of available channels or the number of the sink's children.
- It is important to build a routing tree where no subtree rooted at a sink's child has a slot request considerably higher than the average. Indeed, a data gathering tree where all subtrees are balanced in terms of slot requests would require fewer slots to ensure data gathering.

*Property 5:* In a raw data convergecast, the maximum packet delivery time, denoted  $MaxDelivery$  is given by Equation 7

$$MaxDelivery \leq (Reprod+1) \times SlotframeSize \times SlotDuration. \quad (7)$$

*Proof :* Let us consider a Multislotframe, where each Data Slotframe reproduces every  $Reprod$  Slotframes. Let us consider any sensor node  $i$  and any packet  $p$  generated by this node. The worst delivery time is obtained for  $p$  when it is generated just after the last slot granted to  $i$  has just started. Packet  $p$  cannot be transmitted in this Data Slotframe. It has to wait the next Data Slotframe that occurs  $Reprod$  Slotframes later. Since according to Assumption A7, any packet transmitted in a given Data Slotframe is delivered to the sink in the same Slotframe,  $p$  experienced at most one additional Slotframe to be delivered to the sink. Hence, the maximum delivery time of packet  $p$  is equal to  $(Reprod+1) \times SlotframeSize \times SlotDuration$ .

**Notice that Equation 7 justifies the choice of a schedule minimizing the total number of slots.**

*Corollary 1:* The Slotframe size should meet:

$$SlotframeSize \leq \frac{L}{(Reprod+1) \times SlotDuration} \quad (8)$$

Assuming  $Reprod = 2$  and  $SlotDuration = 10ms$ , which are the default values, Equation 8 gives for  $L = 1200ms$ ,  $SlotframeSize \leq 40$  slots.

### C. Feasibility conditions with and without spatial reuse

Without spatial reuse, each cell defined by the couple  $(slotOffset, channelOffset)$  represents an opportunity to transmit for a single node. As a consequence, the number of cells is equal to the number of transmissions. We have the following property:

*Property 6:* Without spatial reuse, the SlotframeSize should meet the following condition, in addition to the conditions expressed in Property 4,

$$SlotframeSize \geq \lceil \frac{\sum_i Gen(i) * Depth(i)}{NChannel} \rceil + \delta_{noreuse} \quad (9)$$

with  $\delta_{noreuse} = 1$  if  $\sum_i Gen(i) * Depth(i)$  is multiple of  $NChannel$  and  $\min(Nchild, NInterf) < NChannel$ ,  $\delta_{noreuse} = 0$  otherwise.

*Proof:* Without spatial reuse, the SlotframeSize should contain a number of cells at least equal to the total number of transmissions. The term  $\delta_{noreuse}$  has been

introduced to deal with topologies where the total number of transmissions is a multiple of  $NChannel$  and  $\min(Nchild, NInterf) < NChannel$ . In such a case, we observe that in the last slot all transmissions are mandatorily transmissions from the children of the sink toward the sink. At most  $\min(Nchild, NInterf) < Nchannel$  transmissions are possible in the same slot. Hence a contradiction with the fact that all channels are used in this slot since the total number of transmissions is a multiple of  $NChannel$ . Hence, an additional slot is needed.

With spatial reuse, several transmissions can share the same cell without interfering. All the difficulty lies in determining whether two nodes conflict or not. This requires a perfect knowledge of the 1-hop and 2-hop neighborhoods as well as a partial knowledge of the 3-hop neighborhood. This requires messages exchanges that we wish to avoid. In addition a schedule without spatial reuse is more robust than a schedule with spatial reuse: it remains valid even if new wireless links appear. However, spatial reuse may be required to meet the latency constraint given by the application.

We are now able to determine the size of the Slotframe taking into account various constraints: latency constraint with Property 5, robustness constraint with Property 1, raw data convergecast with Property 4, no spatial reuse with Property 6.

The feasibility conditions without spatial reuse become:

$$SlotframeSize \geq \max\left(\frac{\sum_i Gen(i)}{\min(NChannel, NInterf, NChild)} + \delta_n, \right. \\ \left. Gen(c_1) + 2 \sum_{\substack{v \in subtree(c_1) \\ v \neq c_1}} Gen(v) + \delta_t, \right. \\ \left. \left\lceil \frac{\sum_i Gen(i) * Depth(i)}{NChannel} \right\rceil + \delta_{noreuse} \right) \\ SlotframeSize \leq \frac{L}{(Reprod + 1) \times SlotDuration} \\ SlotframeSize \text{ and } NChannel \text{ should be coprime.}$$

These feasibility conditions are necessary conditions. They give the minimum number of slots needed by data gathering able to meet the latency constraint when the problem is feasible without spatial reuse. The next step will be to compute a valid schedule that meets this minimum number of slots.

#### IV. A DEBT-BASED SCHEDULE

##### A. Characterization of the schedule

To characterize the schedule we want to compute, we use the following concepts.

**Definition 1:** A valid schedule is a schedule where in any timeslot of the schedule:

- the number of interfaces used by each wireless node, taking into account all the channelOffsets used, is less than or equal to the number of interfaces of this node;
- at most,  $NChannels$  are used;
- a transmission is scheduled if and only if the transmitter has at least one packet to transmit, and it has at least one interface available as well as its parent.

**Definition 2:** A schedule is said to be *collision-free* if and only if no two conflicting nodes are assigned the same timeslot and the same channel.

**Definition 3:** A schedule is said to be *traffic-aware* if and only if each sensor node is assigned a number of slots that enables it to transmit all its messages in the same Slotframe.

**Definition 4:** A traffic-aware collision-free schedule is said to *minimize data gathering delays* if and only if the total number of slots needed to allow each sensor node to transmit all its messages in the same Slotframe and deliver them to the sink in the same Slotframe, is minimized.

**We want to compute a valid schedule that is collision-free, traffic-aware and minimizes the data gathering delays.** Among all the possible schedulers, we select the debt-based schedulers as developed in the next section.

##### B. A simple Debt-based scheduler

**Definition 5:** A scheduler is said to be *debt-based* if and only if it schedules first the node having the highest debt.

In our context, we propose to define the debt of a node  $i$  as follows:

$Debt(i) = RemainTrans(i) \times Depth(i)$  if node  $i$  has at least one packet in its Transmit queue, and 0 otherwise.

$Depth(i)$  is the distance of  $i$  to the sink expressed in the number of hops, and  $RemainTrans(i)$  is the number of packets remaining to be scheduled, even if only one of them is present in the Transmit queue.

The debt of a node  $i$  is updated when:

- a packet of node  $i$  is transmitted by  $i$ ;
- a packet of a child of  $i$  is received by  $i$ ;
- a new packet is generated on node  $i$ .

We focus on a simple debt-based scheduler that schedules at most one transmission per cell. Each cell is defined by its couple  $(slotOffset, channelOffset)$ . In other words, this scheduler makes no spatial reuse of the timeslots. This scheduler orders the nodes by decreasing debt. It selects the node with the highest debt and inserts a transmission of this node to its parent in the current timeslot and current channeloffset if and only if this node and its parent have an available interface in this slot. Otherwise, the node with the second highest debt is tried, and so on. The scheduler then moves to the next channelOffset. If all channel offsets have been visited, it moves to the next timeslot, until all the packets generated by the sensor nodes have been received by the sink. This debt-based scheduler is used to compute the schedules without spatial reuse for all the scenarii studied in Section V.

#### V. PERFORMANCE EVALUATION

Our goal is to evaluate the performance of different scenarii running on a random network topology. The performance evaluation is conducted using the NS3 simulator [9], that we extend to integrate the functionalities needed to support our framework including the multi-interface management.

### A. The application considered

We focus on a network consisting of 20 sensor nodes plus the sink denoted as node 0. The topology of this network is randomly selected. The routing tree associated with this topology is depicted in Figure 2. It has a depth of three. Four sensor nodes are one-hop away from the sink. These nodes have ten children that have six children as depicted in Figure 2. Such a configuration is representative of the industrial application considered. The sampling rate of each sensor node is given in Table III.

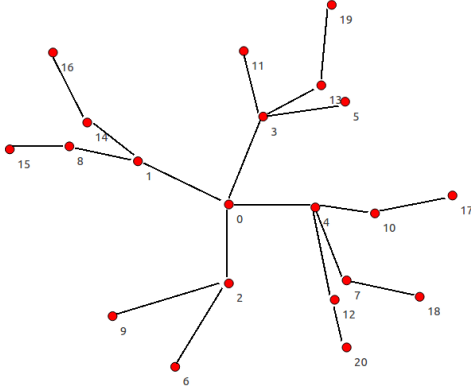


Fig. 2: An example of a 3-hop random network topology.

TABLE III: Sampling rate of each sensor node.

Sampling rate per second	Sensor nodes
400	1 2
300	3 4 5 6 9 15 19
100	7 8 10 11 12 13 14 16 17 18 20

### B. Feasibility conditions for this configuration

For this configuration where the sink has 4 children, the feasibility conditions established in Section III-C are given in Table IV. They show the impact of the numbers of interfaces and channels on the number of slots needed by data gathering for the topology considered. For the feasibility conditions with spatial reuse, we assume that the only topology links are those of the data gathering tree. With and without spatial reuse, we observe that **an increase in the number of interfaces of the sink strictly higher than the number of sink's children brings no benefit**. Similarly, **a number of interfaces higher than the number of channels has no interest**. An increase in the number of channels or interfaces that increases the value of  $\min(N_{channel}, N_{child}, N_{interf})$  tends to reduce the number of slots needed or keeps the same value in the worst case. This is explained by Property 4.

For 4 channels, the number of slots obtained in Table IV does not take into account the constraint expressing that  $SlotframeSize$  and  $N_{channel}$  should be coprime to avoid that the same transmission is done on the same physical channel in two consecutive Data Slotframes. If this constraint is taken into account, the minimum number of slots would be without reuse, 21 slots for 2 and 3 interfaces, and 15 slots for 4 interfaces; whereas with reuse it would be 21 slots for 2 interfaces, and 15 slots for 3 and 4 interfaces.

As a consequence, selecting an even number of channels for a TSCH network is interesting only if the perturbations of channels are very short (i.e. less than  $Reprod \times SlotframeSize \times SlotDuration$ ). Otherwise, an even number of channels may lead to a larger  $SlotframeSize$  that is not compatible with the short latency required by the application, as illustrated by this example, where we obtain 21 slots for 4 channels instead of 16 slots for 3 channels. **No valid schedule may need a smaller number of slots than this given in Table IV for the configuration considered.**

TABLE IV: Minimum number of slots needed for various numbers of channels and interfaces.

# Int.	3 channels		4 channels		5 channels	
	No reuse	Reuse	No reuse	Reuse	No reuse	Reuse
1	Unfeas.	Unfeas.	Unfeas.	Unfeas.	Unfeas.	Unfeas.
2	Unfeas.	16 slots	16 slots	16 slots	16 slots	16 slots
3	Unfeas.	14 slots	16 slots	14 slots	14 slots	14 slots
4	Unfeas.	14 slots	15 slots	14 slots	14 slots	14 slots

### C. Simulation results

We now consider different scenarii, corresponding to different numbers of channels and sink's interfaces. For each scenario, we trigger 30 simulation runs. Each simulation run differs by the random generation times of the first packet on the sensor nodes. Simulation parameters are given in Table V.

TABLE V: Simulation parameters.

Parameter	Value
$N_{Channel}$	4, 5
$N_{Interf}$	4, 3, 2 or 1
$N_{Child}$	4
$Latency$	1200 ms
$SlotDuration$	10 ms
$SlotFrameSize$	variable
$Reprod$	2
Simulation duration	300 s
Number of simulations	30

We evaluate the average and maximum delivery times of packets generated by each sensor node. The packet delivery time is equal to the time where the packet is inserted in the Receive queue at the Network layer of the sink minus the generation time of the packet at the Network layer of the sensor node. It is interesting to compare the maximum delivery time obtained in the simulations with the theoretical one. We also evaluate the average and maximum sizes of the Transmit queue on each sensor node. This size is expressed in terms of the number of packets in this queue. We study the impact of the numbers of channels and sink's interfaces on the packet delivery times and the Transmit queue size.

We first assume that we have 4 channels and the sink has 2 or 3 interfaces. Figure 3 depicts the average and maximum delivery times of packets generated by each sensor node. Nodes 1 and 2 have 4 packets to transmit in each Slotframe. We can see that these nodes have the smallest packet delivery delay. Nodes 3, 4, 5, 6, 9, 15 and 19 have 3 packets to transmit in each Slotframe. These nodes have a packet delivery delay less than the remaining nodes that have only 1 packet to transmit per Slotframe. This is due to the debt-based schedule that favors nodes with a high debt.



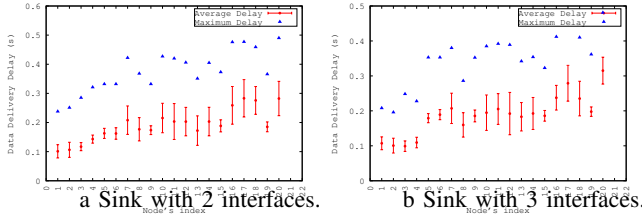


Fig. 3: Average and max. packet delivery times for 4 channels.

Figure 4 depicts the average and maximum delivery times of packets when the number of channels is 5 and the number of sink's interfaces varies from 2 to 4. Compared with Figure 3, the maximum delivery time is smaller when the number of channels increases. When the sink has 2 interfaces, the maximum delivery time reaches  $0.48s$  for 4 channels whereas it decreases to  $0.37s$  with 5 channels. And when the number of sink's interfaces is 3, the maximum delivery time is equal to  $0.48s$  with 4 channels and  $0.34s$  with 5 channels. This decrease in the delivery delay is expected due to Property 4.

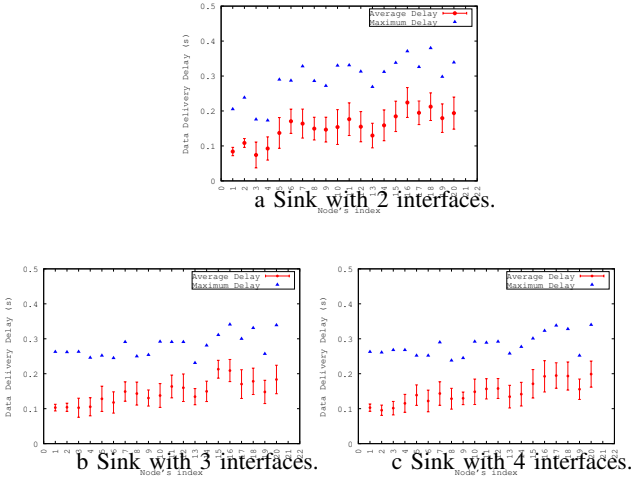


Fig. 4: Average and max. packet delivery times for 5 channels.

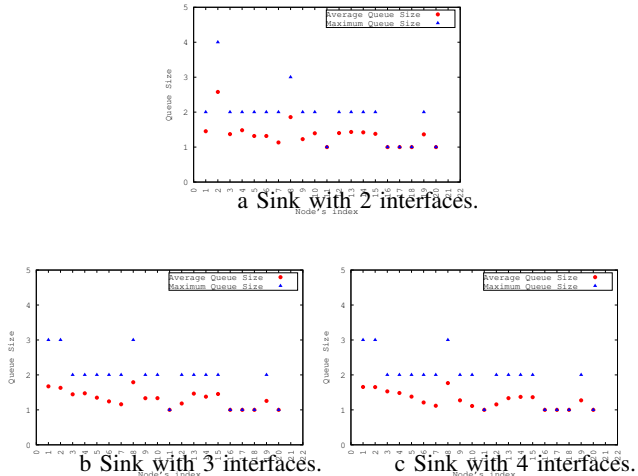


Fig. 5: Size of the Transmit queue for 5 channels.

Figure 5 depicts the average and maximum sizes of the Transmit queue in each sensor node when the number of channels used is equal to 5 and the sink's interfaces varies from 2 to 4. We observe that parent nodes have longer queues than non-parent nodes. This is explained by the fact that they have to forward the packets received from their children. In any case, the maximum size of the queue is 4.

## VI. CONCLUSION

TSCH with its time-slotted and multichannel medium access provides an efficient support for data gathering. In this paper, we showed how applications with latency constraints can be supported by a TSCH network. We established the feasibility conditions for various configurations taking into account heterogeneous packet generation rates, various numbers of channels and sink's interfaces. We determined the conditions for which an increase in the number of channels or sink's interfaces leads to a shorter data delivery delay. We compared the number of slots needed by data gathering with and without spatial reuse for small configurations. We integrated the proposed framework in the NS3 simulator and evaluated the performance in a random topology for different configurations representative of an industrial application. Simulation results showed that the maximum theoretical delivery delay is never exceeded and the number of messages in the Transmit queue of each sensor node remains small. In addition, the debt-based scheduler builds a valid schedule with the minimum number of slots for the industrial application considered.

## REFERENCES

- [1] D. De Guglielmo and G. Anastasi and A. Seghetti, *Advances onto the Internet of Things*. Springer, 2014, ch. From IEEE 802.15.4 to IEEE 802.15.4e: A Step Towards the Internet of Things.
- [2] S. Chen, T. Sun, J. Yuan, X. Geng, C. Li, S. Ullah, and M. A. Alnuem, "Performance Analysis of IEEE 802.15.4e Time Slotted Channel Hopping for Low-Rate Wireless Networks," *KSII Transactions on Internet and Information Systems*, vol. 7, no. 1, pp. 1–21, 2013.
- [3] T. Watteyne and J. Weiss and L. Doherty and J. Simon, "Industrial IEEE802.15.4e Networks: performance and trade-offs," in *IEEE ICC 2015 SAC Internet of Things*, London, UK, June 2015.
- [4] X. Vilajosana and Q. Wang and F. Chraim and T. Watteyne and T. Chang and K. Pister, "A Realistic Energy Consumption Model for TSCH Networks," *IEEE Sensors Journal*, vol. 14, no. 2, February 2009.
- [5] H. Kurunathan and R. Severino and A. Koubaa and E. Tovar, "Towards Worst-Case Bounds Analysis of the IEEE 802.15.4e," in *RTAS 2016, 22nd IEEE Real-Time and Embedded Technology and Applications Symposium*, Vienna, Austria, 2016.
- [6] "IEEE Standard for Local and metropolitan area networks—Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs)," IEEE, IEEE Std 802.15.4-2011 (Revision of IEEE Std 802.15.4-2006), Sept 2011. [Online]. Available: <http://dx.doi.org/10.1109/ieeestd.2006.232110>
- [7] "IEEE Standard for Local and metropolitan area networks—Part 15.4: Low-Rate Wireless Personal Area Networks (LR-WPANs) Amendment 1: MAC Sublayer," Institute of Electrical and Electronics Engineers (IEEE), IEEE Std 802.15.4e-2012 (Amendment to IEEE Std 802.15.4-2011), April 2012.
- [8] T. H. Mengfan Shan, Song Min Kim and G. Chen, "On minimizing data aggregation delay in low duty cycle sensor networks," in *Proceedings of the IEEE MASS 2016 conference*, Brasilia, Brazil, October 2016, pp. 17–31.
- [9] "NS3." [Online]. Available: <https://www.nsnam.org/>